Predict, then smart optimize with stochastic programming

Rohit Kannan[1], Güzin Bayraksan[2], and Jim Luedtke[1]

[1]Wisconsin Institute for Discovery, University of Wisconsin-Madison

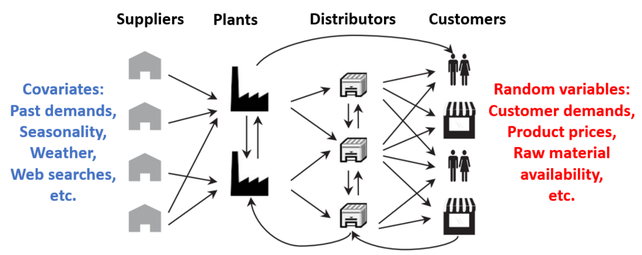
[2]Department of Integrated Systems Engineering, The Ohio State University

Setup

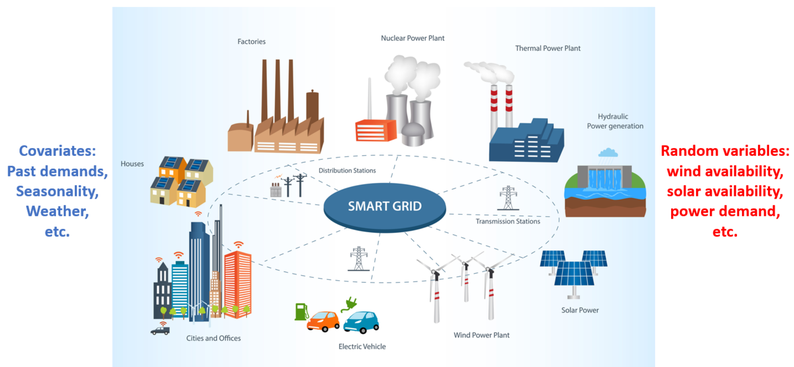
- Traditional SP formulation: $\min_{z \in \mathcal{Z}} \mathbb{E}_Y [c(z, Y)]$
- Data-driven SP: samples $\{y^i\}_{i=1}^n$ of random variables Y
- Often also have data $\{x^i\}_{i=1}^n$ of random features/covariates X that can be used to predict Y

Applications

- Shipment planning under demand uncertainty



- Smart grid operation under renewables uncertainty



Formulation

$$v^*(x) = \min_{z \in \mathcal{Z}} \mathbb{E}[c(z, Y) \mid X = x]$$

- $X = x$ is a new random observation of the covariates
- Concurrent data $\mathcal{D}_n := \{(y^i, x^i)\}_{i=1}^n$ of Y and X
- Let $Y = f^*(X) + \varepsilon$, where $f^*(x) = \mathbb{E}[Y \mid X = x]$
- If we know f^*, can solve $\min_{z \in \mathcal{Z}} \frac{1}{n} \sum_{i=1}^n c(z, f^*(x) + \varepsilon^i)$

Predict-then-smart-optimize frameworks

Learn model to predict Y given $X = x$, use as proxy for f^*. Use the residuals of this model on the training data $\mathcal{D}_n$ as proxy for samples of the errors ε.

- Estimate f^* using the data $\mathcal{D}_n$, e.g.,

$$\hat{f}_n(\cdot) \in \arg\min_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^n \ell\left(y^i, f(x^i)\right)$$

- Use empirical residuals $\hat{\varepsilon}_n^i := y^i - \hat{f}_n(x^i)$ as proxy for samples of ε within a SAA framework

$$\hat{z}_n^{ER}(x) \in \arg\min_{z \in \mathcal{Z}} \frac{1}{n} \sum_{i=1}^n c(z, \hat{f}_n(x) + \hat{\varepsilon}_n^i) \qquad \text{(ER-SAA)}$$

- Using leave-one-out residuals $\hat{\varepsilon}_{n,J}^i := y^i - \hat{f}_{-i}(x^i)$ within the SAA could work better with limited data

$$\hat{z}_n^J(x) \in \arg\min_{z \in \mathcal{Z}} \frac{1}{n} \sum_{i=1}^n c(z, \hat{f}_n(x) + \hat{\varepsilon}_{n,J}^i) \qquad \text{(J-SAA)}$$

Theoretical guarantees

- Conditions on the SP and the learning step for asymptotic optimality, rates of convergence of the ER-SAA, J-SAA solns
  - Applicable to two-stage stochastic linear programs (LPs)
  - Can handle general learning frameworks and time series data $\mathcal{D}_n$
- For ER-SAA, the learning step must satisfy

$$\hat{f}_n(x) \xrightarrow{p} f^*(x) \quad \text{and} \quad \frac{1}{n} \sum_{i=1}^n \|f^*(x^i) - \hat{f}_n(x^i)\|^2 \xrightarrow{p} 0.$$

Setup for computational experiments

Two-stage resource allocation LP model

- Meet demands of 30 customers for 20 resources
- Uncertain demands Y generated according to

$$Y_j = \alpha_j^* + \sum_{l=1}^3 \beta_{jl}^* (X_l)^p + \varepsilon_j, \quad \forall j \in \{1, \cdots, 30\},$$

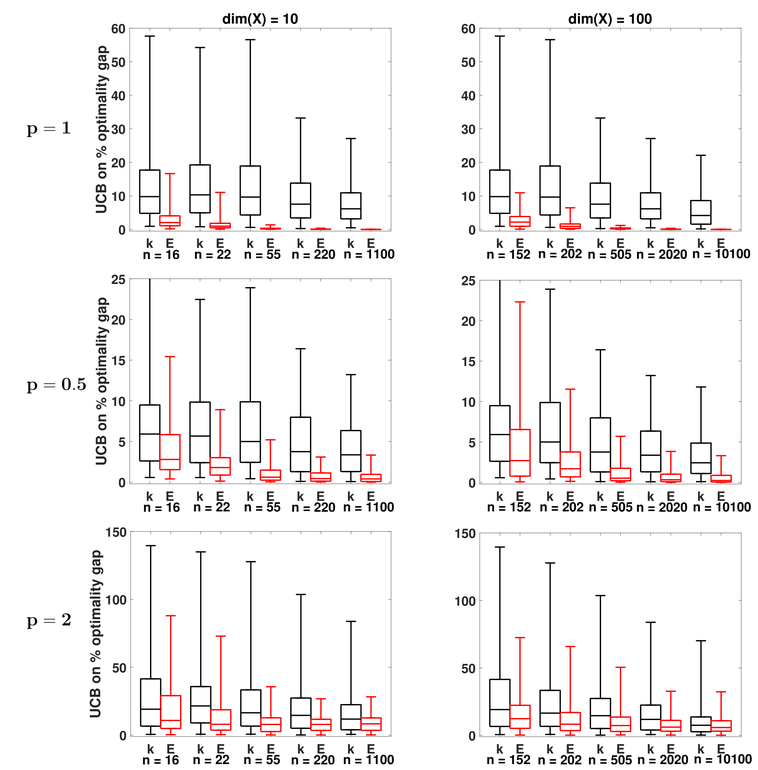
where $\varepsilon_j \sim \mathcal{N}(0, \sigma_j^2)$, $p \in \{0.5, 1, 2\}$, $\dim(X) \in \{10, 100\}$

- Fit a linear model with OLS regression (even when $p \neq 1$)

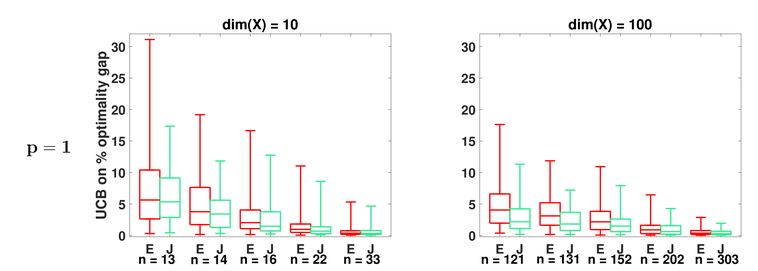
Numerical results

Legend: `k`: kNN-based approach of Bertsimas & Kallus (2019), `E`: ER-SAA + OLS, `J`: J-SAA + OLS, `UCB`: 99% upper confidence bound

- Advantage of using our data-driven formulations



- Advantage of the J-SAA formulation with limited data



Acknowledgment: Funding from the Department of Energy, Office of Science, Office of Advanced Scientific Computing Research, Applied Mathematics program under Contract Number DE-AC02-06CH11347.